INT J COMPUT COMMUN, ISSN 1841-9836
9(1):71-78, February, 2014.

Energy Efficient Key Management Scheme for Wireless Sensor Networks

N. Suganthi, V. Sumathy

N.Suganthi*
Dept of Information Technology
Kumaraguru College of Technology, Coimbatore-49
*Corresponding author suganthiduraisamy@yahoo.co.in

V.Sumathy
ECE Department
Government College of Technology, Coimbatore-13
sumi_gct2001@yahoo.co.in

Abstract:
Designing an efficient key establishment scheme is of great importance to the data security in Wireless Sensor Networks. The traditional cryptographic techniques are impractical in Wireless Sensor Networks because of associated high energy and computational overheads. This algorithm supports the establishment of three types of keys for each sensor node, an individual key shared with the base station, a pair wise key shared with neighbor sensor node, and a group key that is shared by all the nodes in the network. The algorithm used for establishing and updating these keys are energy efficient and minimizes the involvement of the base station. Polynomial function is used in the study to calculate the keys during initialization, membership change and key compromise. Periodically the key will be updated. To overcome the problem of energy insufficiency and memory storage and to provide adequate security, the energy efficient scheme is proposed. It works well in undefined deployment environment. Unauthorized nodes should not be allowed to establish communication with network nodes. This scheme when compared with other existing schemes has a very low overhead in computation, communication and storage.
Keywords: key management, sensor nodes, polynomial function

1 Introduction

These tiny sensor nodes, which consist of sensing, data processing and communicating components, leverage the idea of sensor networks based on the collaborative effort of a large number of nodes. Sensor nodes are deployed in hostile environments or over large geographical area. The nodes could either have a fixed location or could be randomly deployed to monitor the environment. The nodes then sense environmental changes and report them to other nodes over flexible network architecture. They have thus found application domains in battlefield communication, homeland security, pollution sensing and traffic monitoring. The limited factors of using sensor nodes are that they have limited battery power and less memory capacity. To control information access in a sensor environment only authorized node must know the key to disseminate the information that is unknown to the compromised nodes. The communication keys may be pair wise [7],Chan,Du or group wise [1], these keys to be updated to maintain security and resilience to attacks. Some of the proposed work was based on static schemes [7]Liu and some are on dynamic schemes [1]Eltoweissy Though many protocols have been designed for the purpose of security in sensor environment, unfortunately, node compromising is rarely or not enough investigated and most of these protocols have a weak resilience to attack [13].

In this paper, we propose a key management scheme for WSNs in which the pair wise keys and the group wise key are set up through the broadcast information during the network initialization

phase and no further message exchange is needed afterwards. Consequently, the communication overhead is very low. Therefore, the compromise of some sensor nodes will not affect any other non-compromised pair wise keys. For the establishment of keys for new nodes, we propose a composite mechanism based on an algorithm in which resource consumption can also be kept very low and also the transmission of information. Here only the polynomial identifier needs to be communicated to the nodes for establishing group key and pair wise key.

The rest of this paper is organized as follows. In Section 2, some related work and their drawbacks are discussed. In Section 3, energy-efficient key management scheme, in Sections 4 and 5, the security and performance of energy efficient key management scheme are analyzed. Finally, Section 6 deals with the conclusion.

2 Related Works

Many pair wise key distribution schemes [5] [7] [8] [9] have been developed for peer to peer wireless sensor networks and heterogeneous network [6] [12] [14].

In one of the hierarchical schemes [1], the base node calculates the group key using partial keys in bottom up fashion. The partial key of the child node is generated using random number and which is passed to its parent to calculate its partial key which further goes in bottom up fashion finally to calculate the group key. The partial keys are calculated by using a function. The function is expressed as

$$f(k_1, k_2) = \alpha^{k_1+k_2} mod\ p \tag{1}$$

p is the prime number, k1, k2 are the partial keys

The decision for choosing a number of partial keys is based on the key size for the security requirements and the corresponding energy consumption. To guarantee that all the nodes in a group received the information, they send the reply (REP) message. If the cluster head does not get the (REP) from all the node, it re-broadcasts.

When a new node joins the group, the group key is recalculated and again the cluster head broadcasts the newly created group key to all the nodes in the group. The same is repeated when a node leaves the group. This makes the old node, which is deleted, not to know the new key that is created. Also communication takes places between two cluster heads. Due to poor memory capacity and low power of sensor nodes it will be difficult to store all the partial keys and the communication becomes costly as it needs to broadcast the group key once it is created and changed. In our energy efficient key management protocol scheme the group key need not be broadcasted each time.

A tree based key management protocol [2] in which each sensor node is pre-deployed with three keys. One of the keys is used for initial communication i.e for key exchange and tree spanning. After the tree is spanned, this key is deleted from the memory of the sensor node. Then for further communication, the remaining two keys are used. One of these keys is symmetric and is used to encrypt (or decrypt) information sent from the child to the parent. The third key is also symmetric and is used to encrypt (or decrypt) information sent from the parent to the child. The two keys are used to make the task of cryptanalysis attacker difficult. The disadvantage of this scheme is that an attacker gaining access (physical) to the sensor node can obtain the information. Xing Zhang et al [11] proposed an energy efficient distributed deterministic key management protocol (EDDK). Though this scheme provides higher security than the above two schemes namely hierarchical and tree-based protocol, it requires large memory to store data. Sencun Zhu et al [10] proposed a LEAP: efficent security mechanisms for large scale distributed sensor networks and Du et al [11]proposed a scheme using depolyment knowledge.

3 Proposed Scheme: Energy-Efficient Key Management Scheme

To fix the flaws present in the existing key management schemes, we propose an energy efficient key management scheme for WSNs. This scheme mainly focuses on the establishment and maintenance of the pair wise keys as well as the group keys. Unlike hierarchical and tree-based key management schemes, this scheme does not require additional memory for storing the keys after deployment. In this scheme, no key is broadcasted. Each and every node generates the group key and pair wise key using one of the polynomial functions. The polynomial function is identified with the help of its ID. To enhance message security in the network initialization phase, each sensor node makes use of its own individual key. The base station computes the individual key of all the nodes using the unique keys and the IDs that it has stored. This method also enhances security in data transmission with periodic key update. To avoid the replay attack, sequence number is used.

3.1 Overall System Description

The diagram (Figure 1) shows the overall system description. Key will be established after node initialization. If any node joins in the system or a node is compromised, key update will be performed. After detecting the compromised node the keys will be removed from the node memory. The overall system is designed to reduce the computation overhead and it requires less communication between nodes.

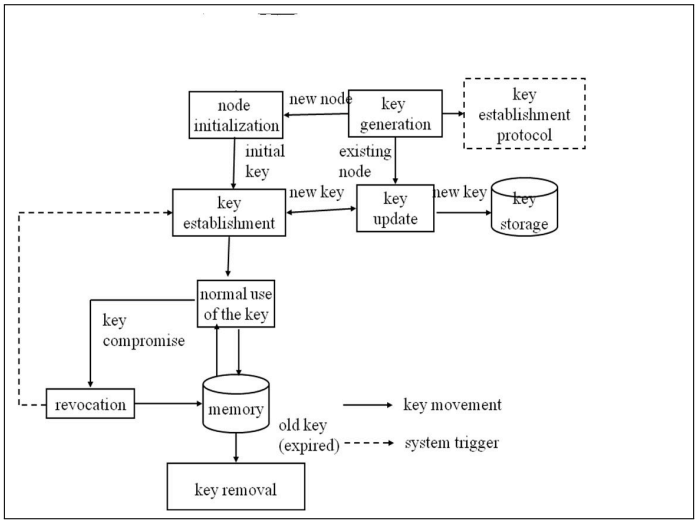

Figure 1: system model

3.2 Spanning of tree

Before deployment, every sensor node is pre-distributed with a network wide shared pseudo-random function (Rf) and an initial key. It is assumed that each node is tamper proof, and it will not be affected by capture attack. The pseudorandom function and the initial key is used by every node in the network to compute its own individual key. This individual key is used for initial communication with the base station. For example, node A s individual key can be computed as per equation (2).

$$K_a = R_f(K_I, Id_a). \tag{2}$$

KI - Initial key, Ida - Individual node identifier

Ka - Individual key shared with base station, Rf -shared pseudorandom function

The hello message which is used to span the tree is also encrypted using this individual key. The hello message contains the ID of the sender and the HELLO keyword. The base station broadcasts hello message. The nodes that reply to hello message become the children of the base station. Here acknowledgment of the child node is essential to accept the node as the child. These nodes then broadcast hello message to other nodes. The nodes that reply to hello message become the children of these nodes. The spanning of the tree is stopped when the nodes do not get a reply to the hello message.

3.3 Key Establishment Phase

Base station will transmit function identifier and random number, encrypted with individual key to individual nodes as shown in equation (3).

$$E_{ka}(Pf_{id}, R_n) \tag{3}$$

Pfid function identifier , Rn random number

Pair Wise key

After sensor nodes are placed in the sensor field, each sensor node communicates with its neighbor via the pair wise key. This key is used to make sure that the message to the intended neighbor node is not known by other neighbor nodes. Nodes A and B are used to show the calculation of the pair wise key using equation (4). Let us consider that node A wants to communicate with node B.

$$K_{ab} = Pf_{id}(R_n, Id_a) \tag{4}$$

Kab Pairwise key, Rn Random number, Ida Identifier of node a

The polynomial function takes the random number and the ID of the node which initiates the communication as input to calculate the pair wise key. The ID of the polynomial function is used to identify one of the functions from the set of polynomial functions. The base node communicates the random number and the ID of the polynomial function to all the nodes. After calculating the pair wise key, it will transmit the message encrypted with this key to the node B along with its ID in plain. Then node B will calculate the pair wise key as it has all the information needed for calculation. Using the key it will decrypt the message transmitted by node A. Here each node, thus, calculates the pair wise key by knowing initiator ID. As no transmission of ID or key information takes place, communication overhead is avoided here.

Group key

A group key is a key shared by all the nodes in the network, and it is needed when the base station is distributing a secure message, (e.g. a query on some event of interest or a confidential instruction) to all the sensor nodes in the network. In a conventional method the parent encrypts M with its cluster key and then broadcasts the message. Each neighbor receiving the message decrypts it to get the message and re-encrypts with its own cluster key, and then transmits the M. This process is repeated until all the nodes receive the message. However, this method has a drawback. In this method, every node has to encrypt and decrypt the message, thus consuming a large amount of energy on computation. So encryption using group key is the most desirable one from the performance point of view. The simple way to store the group key for a node is to preload every node with it. An important problem that arises immediately is the secured

updation of this key when a compromised node is detected. In our proposed scheme, to enable base station-individual nodes communication, group key is used. The group key generator that is present in all the nodes is used for generating the group key using the equation(5). The random number that is transmitted to each node is also involved in key calculation. The group key is calculated as follows.

$$K_g = Pf_{id}(R_n, G_k) \tag{5}$$

Kg - group key , Pfid - polynomial function , Rn - random number, Gk - group key generator

The timer is set for the reestablishment of the key. When the timer reaches the threshold value that is assigned, the re-keying is done. Here re-keying is done by changing the coefficient of the polynomial function. The previously calculated keys are deleted periodically. Therefore, even if an adversary could compromise some legitimate nodes, it still could not compute the pair wise keys and the group key. Note that each sensor node only needs to broadcast one communication message during the key establishment phase with no further message exchange required for key calculation. Thus, the communication overhead can be very low. During the data transfer phase the sequence number is used to indicate the message transfer between the nodes. Once the sequence number reaches the threshold value that is already set, the sequence number is reset to 1 again and it can prevent the replay attacks. This sequence number helps to know the number of the messages sent and received. It is also used for receiving the acknowledgement.

3.4 Key Update Phase

Pair wise key and group key should be updated to avoid cryptanalysis and to prevent attacks from adversaries after one or more sensor nodes are compromised. And also after the threshold time, nodes need to update the keys using the same formula as mentioned in equation (4) and (5). The coefficient of the polynomial function is changed. It is done by adding the constant with the previous values and the modulus is taken to be used as new coefficients. Thus it makes the key update easily and avoids communication overhead.

4 System Analysis

For system analysis, we have implemented the key management algorithm in matlab. Compared with the EDDK, Tree based protocol and hierarchical scheme.

4.1 Computation Costs

Computation costs are measured in terms of number of encryptions required to change the keys in the event of node compromise and node addition. When a node is added, only the new random number and new id of the polynomial function will be transmitted by base station. Individual nodes will receive it and compute the group key and pair wise key using the formulas. Calculation using the polynomial function will consume less energy. But in other schemes lot of encryption and decryption is involved to get the key update. It consumes lot of battery energy. The comparison is made between the existing schemes and the proposed scheme in terms of time. This graph (Figure 2) shows the computation time differences between various schemes and proposed scheme depending upon the number of nodes during initial key calculation.

4.2 Memory Requirement for Key Storage

Let x represent the number of neighboring nodes around a sensor and n be the number of polynomial functions. Each sensor node has x storage units for the pair wise keys, nt + n

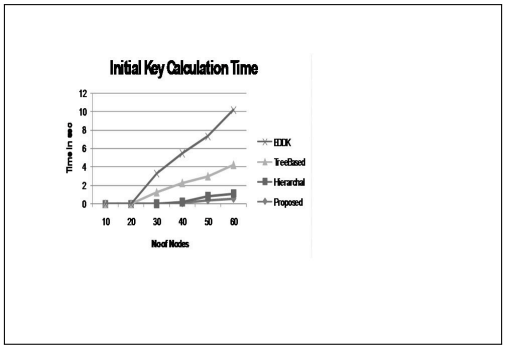

Figure 2: Time required to calculate initial key

storage units for the t-degree polynomial functions, two storage units for random number and pseudo random function and single storage unit to store the group key. In terms of memory requirement to store keys for each scheme, the proposed scheme needs less memory, hence it provides scalability. Even when the number of nodes increases, the memory required to store the keys remains the same. On the contrary the tree based protocol requires more memory as the number of nodes increases. The following graph (Figure 3) shows the comparison of memory requirement for all schemes. The proposed scheme need less memory even when number of nodes increases.

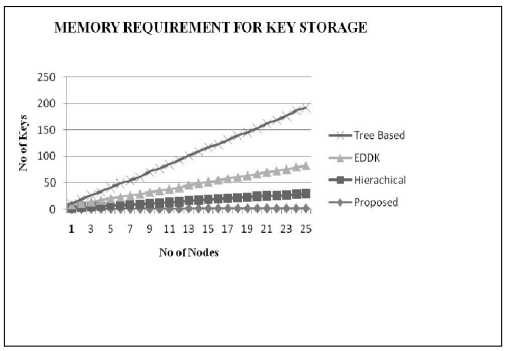

Figure 3: Memory requirement for key storage

4.3 Communication Overhead For Key Exchange

Communication cost is measured in terms of number of messages needed to be exchanged in order to update the existing keys as a result of events like; addition of new node, node compromise, and key refreshing at regular intervals. The communication overhead for the existing scheme is more when compared to the proposed scheme. This is because they need to exchange the keys to enable communication. The proposed scheme requires transmitting the IDs of the polynomial function and random numbers only. So overhead for key exchange is minimal.

As shown in Figure 4, the number of nodes increases, the communication overhead increases for tree based protocol; whereas EDDK and proposed scheme require less communication overhead.

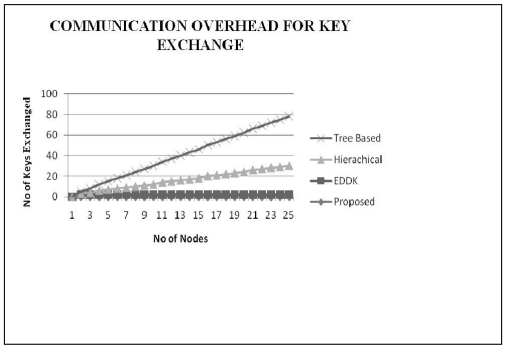

Figure 4: Communication overhead for key exchange

5 Security Analysis

Adversaries with a single compromised node and adversaries with n compromised nodes are chosen selectively. In all cases we study their impact on the desired network properties assuming that the adversary acts maliciously at different layers of the communication protocols. Insiders are adversaries that can compromise nodes or otherwise have a valid identity in a network with appropriate key material. Insiders therefore have the same capabilities as outsiders plus the ability to participate in the network protocols and deviate from the normal behavior of the protocols. Stronger security considerations have to be taken into account for insiders. A minimum level of fault tolerance has to be designed into the network inside attackers. But in our proposed algorithm every node is loaded with the set of polynomials, and every time one function will be used to calculate the key. Adversary nodes cannot generate this polynomial functions. And also it doesn't know which function will be used to calculate the key at that time. The random number will be communicated to the individual nodes by the base station after encrypting with secret key shared by the base station and the individual node.

6 Conclusion

The key exchange problem for sensor networks has been introduced and it is believed that information can be secured by not exchanging the keys directly. A mechanism that makes use of pre-deployed functions has been proposed to fulfill our idea. By using this mechanism, the impacts of many attacks in wireless sensor networks can be limited. This scheme incorporates mechanism that allows for the scalability of memory in the sensor nodes. By comparing the proposed scheme with the existing scheme, it becomes clear that memory required to store the key information is less. Similarly the communication overhead to exchange the keys is also low. Because of the increase in the complexity of the algorithm, the immunity of the sensor networks towards various attacks has been greatly increased. Thus the proposed naming mechanism shields the network from various attacks.

Bibliography

[1] Biswajit Panja; Sanjay Madria; Bharat Bhargava; Energy-Efficient Group Key Management Protocols for Hierarchical Sensor Networks,*Int. J. of Distributed Sensor Networks Taylor Francis Group*, 201-223, DOI:10.1080/15501320701205225, 2007.

[2] Messai,L.; Aliouat,M.; Seba,H.; Tree Based Protocol for Key Management in Wireless Sensor Networks, *EURASIP J.on Wireless Communications and Networking*, Article ID 910695, DOI:10.1155/2010/910695, 2010.

[3] Xing Zhang; Jingsha He; QianWei; EDDK: Energy-Efficient Distributed Deterministic Key Management for Wireless Sensor Networks,*EURASIP J. on Wireless Communications and Networking*, Article ID 765143, DOI:10.1155/2011/765143, 2011.

[4] Eltoweissy,M.; Moharrum,M.; Mukkamala,R.; Dynamic key management in sensor networks, *IEEE Communications Magazine*, 44(4):122- 130, 2006.

[5] Du,W.; Deng,J.; Han,Y.S.; Varshney,P.K.; Katz,J.; Khalili,A.; A pairwise key predistribution scheme for wireless sensor networks, *ACM Trans. on Information and System Security*, 8(2):228-258, 2005.

[6] Jen Yan Huang; I-En Liao; Hao-Wen Tang; A Forward Authentication Key Management Scheme for Heterogeneous Sensor Networks, *EURASIP J. on Wireless Communications and Networking*, Article ID 296704, DOI:10.1155/2011/296704, 2011.

[7] Eschenauer,L.; Gligor,V.D.(2002); A key-management scheme for distributed sensor networks, *Proc. of the 9th ACM Conference on Computer and Communications Security*, Washington, DC, USA, 41-47, 2002.

[8] Chan,H.; Perrig,A.; Song, D.; Randomkey predistribution schemes for sensor networks, *Proc. of IEEE Symposium on Security And Privacy*, 197-213, 2003.

[9] Liu,D.; Ning, P.; Establishing pairwise keys in distributed sensor networks,*Proc. of the 10th ACM Conference on Computer and Communications Security (CCS 03)*, Washington, DC, USA, 52-61, 2003.

[10] Sencun Zhu; Sanjeev Setia; Sushil Jajodia (2003); LEAP: Efficient Security Mechanisms for Large Scale Distributed Sensor Networks, *Proc. of the 10th ACM Conference on Computer and Communications Security*, pp.62-72.

[11] Du,W.; Deng,J.; Han,Y.S.; Chen,S.; Varshney,P.; A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge, *Proc. IEEE INFOCOM04*, 586-597, 2004.

[12] Kausar,F.; Hussain,S.; Yang,L.T.; Masood,A.; Scalable and efficient key management for heterogeneous sensor networks, *J. of Supercomputing*, 45(1):44-65, 2008.

[13] Xiao,Y.; Rayi,V.K.; Sun,B.; Du,X.; Hu,F.; Galloway,M.; A survey of key management schemes in wireless sensor networks,*J. of Computers Communications*, 30(11-12):2314-2341, 2007.

[14] Du,X.; Xiao, Y.; Guizani, M.; Chen,H.H.; An effective key management scheme for heterogeneous sensor networks, *J. of Ad Hoc Networks*, 5(1):24-34, 2007.